

TALLER HERENCIA Y POLIMORFISMO

DANIEL SALAZAR LOAIZA

ADSO - 2873707

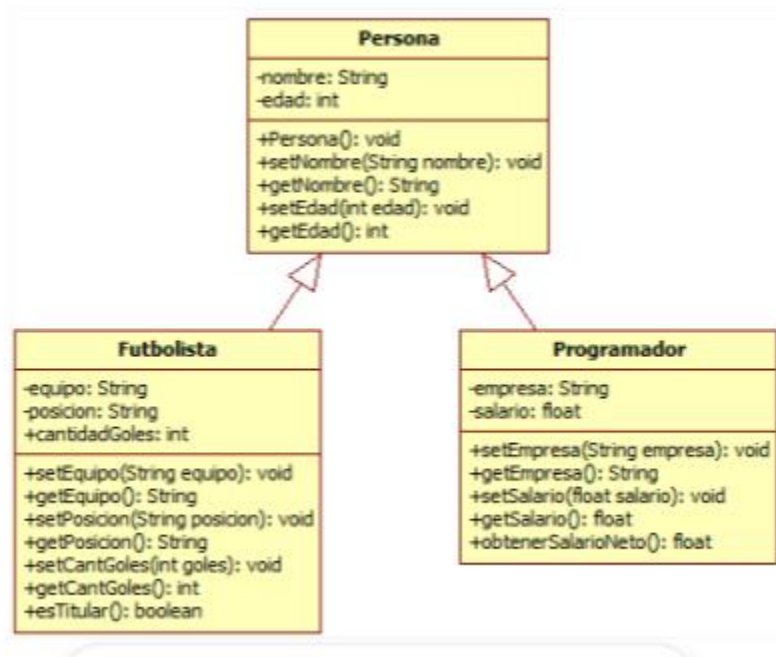
CENTRO DE PROCESOS INDUSTRIALES Y CONSTRUCCION (CPIC)

REGIONAL CALDAS

12 de septiembre del 2024

TALLER HERENCIA Y POLIMORFISMO EN DART - 01

1. Crear un proyecto en Dart y seguir las siguientes instrucciones, tomando como base el siguiente diagrama de clases:



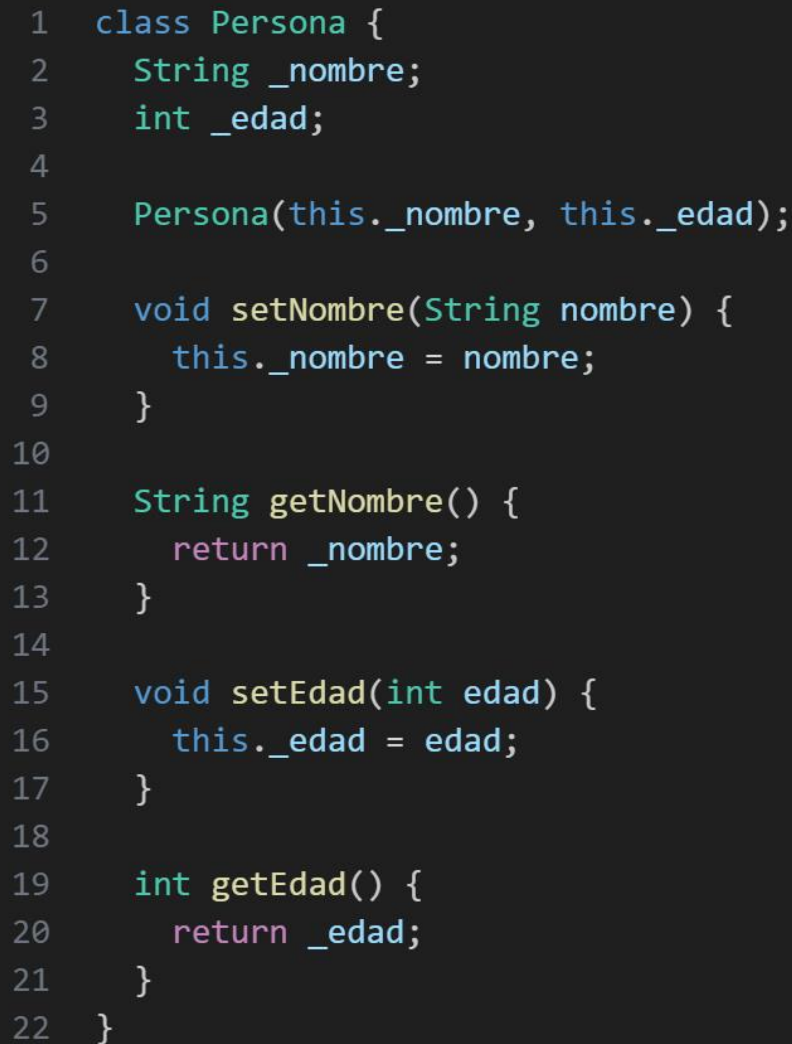
- A.** Construir las 3 clases en Dart tomando como base el diagrama de clases anterior.
- B.** Para llamar el constructor de la clase padre desde la clase Futbolista y Programador, se deberá hacer utilizando el método correspondiente de la herencia.
- C.** Crear 1 objeto (instancia) de la clase Persona, 2 objetos (instancias) de la clase Futbolista y 2 objetos (instancias) de la clase Programador.
- D.** El método `esTitular` de la clase Futbolista, será un método que devolverá un valor booleano que se determinará de la siguiente manera: Si el futbolista tiene más de 5 goles será titular del equipo, si tiene menos no lo podrá ser. El sistema deberá mostrar todos los datos del futbolista y si será titular o no. Tomar un pantallazo de los resultados.
- E.** El método `obtenerSalarioNeto` de la clase Programador, será un método que devolverá el salario neto de un programador, este valor será calculado de la siguiente manera: al salario actual se le restará el 11% del salario. El sistema deberá mostrar todos los datos del programador y el salario neto. Tomar un pantallazo de los resultados.

SOLUCION

En este archivo de la parte uno cree cuatro paginas llamadas “Futbolista.dart”, “Programador.dart”, “Persona.dart” y “Main.dart”.

CODIGOS:

Persona.dart:



```
1  class Persona {
2    String _nombre;
3    int _edad;
4
5    Persona(this._nombre, this._edad);
6
7    void setNombre(String nombre) {
8      this._nombre = nombre;
9    }
10
11   String getNombre() {
12     return _nombre;
13   }
14
15   void setEdad(int edad) {
16     this._edad = edad;
17   }
18
19   int getEdad() {
20     return _edad;
21   }
22 }
```

Futbolista.dart:

```
1 import 'Persona.dart';
2
3 class Futbolista extends Persona{
4     String _equipo;
5     String _posicion;
6     int cantidadGoles;
7
8     Futbolista(this._equipo, this._posicion, this.cantidadGoles, String _nombre, int _edad): super(_nombre, _edad);
9
10    void setEquipo(String equipo) {
11        _equipo = equipo;
12    }
13
14    String getEquipo() {
15        return _equipo;
16    }
17
18    void setPosicion(String posicion) {
19        _posicion = posicion;
20    }
21
22    String getPosicion() {
23        return _posicion;
24    }
25
26    void setCantGoles(int goles) {
27        cantidadGoles = goles;
28    }
29
30    int getCantGoles() {
31        return cantidadGoles;
32    }
33
34    bool esTitular() {
35        if (cantidadGoles > 5) {
36            return true;
37        } else {
38            return false;
39        }
40    }
41
42    @override
43    String getNombre() {
44        return super.getNombre();
45    }
46
47    @override
48    int getEdad() {
49        return super.getEdad();
50    }
51
52    void mostrarInformacion() {
53        print('*' * 50);
54        print("""
55        nombre: ${getNombre()}
56        edad: ${getEdad()}
57        equipo: ${getEquipo()}
58        posicion: ${getPosicion()}
59        goles: ${getCantGoles()}
60        Titularidad: ${esTitular()}
61        """);
62        print('*' * 50);
63    }
64
65 }
```

Programador.dart:

```
1  import 'Persona.dart';
2
3  class Programador extends Persona {
4    String _empresa;
5    double _salario;
6
7    Programador(this._empresa, this._salario, String _nombre, int _edad): super(_nombre, _edad);
8
9    void setEmpresa(String empresa) {
10     _empresa = empresa;
11   }
12
13   String getEmpresa() {
14     return _empresa;
15   }
16
17   void setSalario(double salario) {
18     _salario = salario;
19   }
20
21   double getSalario() {
22     return _salario;
23   }
24
25   double obtenerSalarioNeto(double salario) {
26     /*salario = _salario * 0.11;
27     _salario = salario;
28     return _salario;*/
29     /*return _salario * 0.11;*/
30     salario = _salario * 0.11;
31     _salario -= salario;
32     return _salario;
33   }
34 }
35
36
37
38 @override
39 String getNombre() {
40   return super.getNombre();
41 }
42
43 @override
44 int getEdad() {
45   return super.getEdad();
46 }
47
48 void mostrarInformacion() {
49   print('* * 50);
50   print("""
51   nombre: ${getNombre()}
52   edad: ${getEdad()}
53   empresa: ${getEmpresa()}
54   salario: ${getSalario()}
55   """);
56   print('* * 50);
57 }
58 }
```

main.dart:

```

1 // Autor: daniel313
2
3 // Importa: Variables globales
4 import { useState } from 'react';
5
6 // Importa: Componentes de React
7 import { BrowserRouter as Router, Route, Switch, Link } from 'react-router-dom';
8
9 // Importa: Componentes de Bootstrap
10 import { Container, Row, Col, Card, Button, Form, InputGroup, FormControl, Table } from 'react-bootstrap';
11
12 // Importa: Componentes de Axios
13 import axios from 'axios';
14
15 // Importa: Componentes de Moment.js
16 import moment from 'moment';
17
18 // Importa: Componentes de Chart.js
19 import { Line } from 'react-chartjs-2';
20
21 // Importa: Componentes de D3.js
22 import { Line } from 'd3.js';
23
24 // Importa: Componentes de Lodash
25 import { debounce } from 'lodash';
26
27 // Importa: Componentes de Redux
28 import { useDispatch, useSelector } from 'react-redux';
29
30 // Importa: Componentes de Redux Toolkit
31 import { createSlice, createReducer, createEntityAdapter, createAsyncThunk } from '@reduxjs/toolkit';
32
33 // Importa: Componentes de RTK Query
34 import { createApi, fetchBaseQuery } from '@reduxjs/toolkit/query';
35
36 // Importa: Componentes de Next.js
37 import { useRouter } from 'next/router';
38
39 // Importa: Componentes de GraphQL
40 import { useQuery, useMutation } from '@apollo/client';
41
42 // Importa: Componentes de Firebase
43 import { auth, db, storage, functions } from 'firebase';
44
45 // Importa: Componentes de Stripe
46 import { StripeCheckout } from 'react-stripe-checkout';
47
48 // Importa: Componentes de PayPal
49 import { PayPalButtons, usePayPalConfig } from '@paypal/react-paypal-js';
50
51 // Importa: Componentes de Twilio
52 import { Twilio } from 'twilio';
53
54 // Importa: Componentes de SendGrid
55 import { SendGrid } from 'sendgrid';
56
57 // Importa: Componentes de Mailgun
58 import { Mailgun } from 'mailgun';
59
60 // Importa: Componentes de Braintree
61 import { Braintree } from 'braintree';
62
63 // Importa: Componentes de Stripe
64 import { Stripe } from 'stripe';
65
66 // Importa: Componentes de Stripe
67 import { Stripe } from 'stripe';
68
69 // Importa: Componentes de Stripe
70 import { Stripe } from 'stripe';
71
72 // Importa: Componentes de Stripe
73 import { Stripe } from 'stripe';
74
75 // Importa: Componentes de Stripe
76 import { Stripe } from 'stripe';
77
78 // Importa: Componentes de Stripe
79 import { Stripe } from 'stripe';
80
81 // Importa: Componentes de Stripe
82 import { Stripe } from 'stripe';
83
84 // Importa: Componentes de Stripe
85 import { Stripe } from 'stripe';
86
87 // Importa: Componentes de Stripe
88 import { Stripe } from 'stripe';
89
90 // Importa: Componentes de Stripe
91 import { Stripe } from 'stripe';
92
93 // Importa: Componentes de Stripe
94 import { Stripe } from 'stripe';
95
96 // Importa: Componentes de Stripe
97 import { Stripe } from 'stripe';
98
99 // Importa: Componentes de Stripe
100 import { Stripe } from 'stripe';
101
102 // Importa: Componentes de Stripe
103 import { Stripe } from 'stripe';
104
105 // Importa: Componentes de Stripe
106 import { Stripe } from 'stripe';
107
108 // Importa: Componentes de Stripe
109 import { Stripe } from 'stripe';
110
111 // Importa: Componentes de Stripe
112 import { Stripe } from 'stripe';
113
114 // Importa: Componentes de Stripe
115 import { Stripe } from 'stripe';
116
117 // Importa: Componentes de Stripe
118 import { Stripe } from 'stripe';
119
120 // Importa: Componentes de Stripe
121 import { Stripe } from 'stripe';
122
123 // Importa: Componentes de Stripe
124 import { Stripe } from 'stripe';
125
126 // Importa: Componentes de Stripe
127 import { Stripe } from 'stripe';
128
129 // Importa: Componentes de Stripe
130 import { Stripe } from 'stripe';
131
132 // Importa: Componentes de Stripe
133 import { Stripe } from 'stripe';
134
135 // Importa: Componentes de Stripe
136 import { Stripe } from 'stripe';
137
138 // Importa: Componentes de Stripe
139 import { Stripe } from 'stripe';
140
141 // Importa: Componentes de Stripe
142 import { Stripe } from 'stripe';
143
144 // Importa: Componentes de Stripe
145 import { Stripe } from 'stripe';
146
147 // Importa: Componentes de Stripe
148 import { Stripe } from 'stripe';
149
150 // Importa: Componentes de Stripe
151 import { Stripe } from 'stripe';
152
153 // Importa: Componentes de Stripe
154 import { Stripe } from 'stripe';
155
156 // Importa: Componentes de Stripe
157 import { Stripe } from 'stripe';
158
159 // Importa: Componentes de Stripe
160 import { Stripe } from 'stripe';
161
162 // Importa: Componentes de Stripe
163 import { Stripe } from 'stripe';
164
165 // Importa: Componentes de Stripe
166 import { Stripe } from 'stripe';
167
168 // Importa: Componentes de Stripe
169 import { Stripe } from 'stripe';
170
171 // Importa: Componentes de Stripe
172 import { Stripe } from 'stripe';
173
174 // Importa: Componentes de Stripe
175 import { Stripe } from 'stripe';
176
177 // Importa: Componentes de Stripe
178 import { Stripe } from 'stripe';
179
180 // Importa: Componentes de Stripe
181 import { Stripe } from 'stripe';
182
183 // Importa: Componentes de Stripe
184 import { Stripe } from 'stripe';
185
186 // Importa: Componentes de Stripe
187 import { Stripe } from 'stripe';
188
189 // Importa: Componentes de Stripe
190 import { Stripe } from 'stripe';
191
192 // Importa: Componentes de Stripe
193 import { Stripe } from 'stripe';
194
195 // Importa: Componentes de Stripe
196 import { Stripe } from 'stripe';
197
198 // Importa: Componentes de Stripe
199 import { Stripe } from 'stripe';
200
201 // Importa: Componentes de Stripe
202 import { Stripe } from 'stripe';
203
204 // Importa: Componentes de Stripe
205 import { Stripe } from 'stripe';
206
207 // Importa: Componentes de Stripe
208 import { Stripe } from 'stripe';
209
210 // Importa: Componentes de Stripe
211 import { Stripe } from 'stripe';
212
213 // Importa: Componentes de Stripe
214 import { Stripe } from 'stripe';
215
216 // Importa: Componentes de Stripe
217 import { Stripe } from 'stripe';
218
219 // Importa: Componentes de Stripe
220 import { Stripe } from 'stripe';
221
222 // Importa: Componentes de Stripe
223 import { Stripe } from 'stripe';
224
225 // Importa: Componentes de Stripe
226 import { Stripe } from 'stripe';
227
228 // Importa: Componentes de Stripe
229 import { Stripe } from 'stripe';
230
231 // Importa: Componentes de Stripe
232 import { Stripe } from 'stripe';
233
234 // Importa: Componentes de Stripe
235 import { Stripe } from 'stripe';
236
237 // Importa: Componentes de Stripe
238 import { Stripe } from 'stripe';
239
240 // Importa: Componentes de Stripe
241 import { Stripe } from 'stripe';
242
243 // Importa: Componentes de Stripe
244 import { Stripe } from 'stripe';
245
246 // Importa: Componentes de Stripe
247 import { Stripe } from 'stripe';
248
249 // Importa: Componentes de Stripe
250 import { Stripe } from 'stripe';
251
252 // Importa: Componentes de Stripe
253 import { Stripe } from 'stripe';
254
255 // Importa: Componentes de Stripe
256 import { Stripe } from 'stripe';
257
258 // Importa: Componentes de Stripe
259 import { Stripe } from 'stripe';
260
261 // Importa: Componentes de Stripe
262 import { Stripe } from 'stripe';
263
264 // Importa: Componentes de Stripe
265 import { Stripe } from 'stripe';
266
267 // Importa: Componentes de Stripe
268 import { Stripe } from 'stripe';
269
270 // Importa: Componentes de Stripe
271 import { Stripe } from 'stripe';
272
273 // Importa: Componentes de Stripe
274 import { Stripe } from 'stripe';
275
276 // Importa: Componentes de Stripe
277 import { Stripe } from 'stripe';
278
279 // Importa: Componentes de Stripe
280 import { Stripe } from 'stripe';
281
282 // Importa: Componentes de Stripe
283 import { Stripe } from 'stripe';
284
285 // Importa: Componentes de Stripe
286 import { Stripe } from 'stripe';
287
288 // Importa: Componentes de Stripe
289 import { Stripe } from 'stripe';
290
291 // Importa: Componentes de Stripe
292 import { Stripe } from 'stripe';
293
294 // Importa: Componentes de Stripe
295 import { Stripe } from 'stripe';
296
297 // Importa: Componentes de Stripe
298 import { Stripe } from 'stripe';
299
300 // Importa: Componentes de Stripe
301 import { Stripe } from 'stripe';
302
303 // Importa: Componentes de Stripe
304 import { Stripe } from 'stripe';
305
306 // Importa: Componentes de Stripe
307 import { Stripe } from 'stripe';
308
309 // Importa: Componentes de Stripe
310 import { Stripe } from 'stripe';
311
312 // Importa: Componentes de Stripe
313 import { Stripe } from 'stripe';
314
315 // Importa: Componentes de Stripe
316 import { Stripe } from 'stripe';
317
318 // Importa: Componentes de Stripe
319 import { Stripe } from 'stripe';
320
321 // Importa: Componentes de Stripe
322 import { Stripe } from 'stripe';
323
324 // Importa: Componentes de Stripe
325 import { Stripe } from 'stripe';
326
327 // Importa: Componentes de Stripe
328 import { Stripe } from 'stripe';
329
330 // Importa: Componentes de Stripe
331 import { Stripe } from 'stripe';
332
333 // Importa: Componentes de Stripe
334 import { Stripe } from 'stripe';
335
336 // Importa: Componentes de Stripe
337 import { Stripe } from 'stripe';
338
339 // Importa: Componentes de Stripe
340 import { Stripe } from 'stripe';
341
342 // Importa: Componentes de Stripe
343 import { Stripe } from 'stripe';
344
345 // Importa: Componentes de Stripe
346 import { Stripe } from 'stripe';
347
348 // Importa: Componentes de Stripe
349 import { Stripe } from 'stripe';
350
351 // Importa: Componentes de Stripe
352 import { Stripe } from 'stripe';
353
354 // Importa: Componentes de Stripe
355 import { Stripe } from 'stripe';
356
357 // Importa: Componentes de Stripe
358 import { Stripe } from 'stripe';
359
360 // Importa: Componentes de Stripe
361 import { Stripe } from 'stripe';
362
363 // Importa: Componentes de Stripe
364 import { Stripe } from 'stripe';
365
366 // Importa: Componentes de Stripe
367 import { Stripe } from 'stripe';
368
369 // Importa: Componentes de Stripe
370 import { Stripe } from 'stripe';
371
372 // Importa: Componentes de Stripe
373 import { Stripe } from 'stripe';
374
375 // Importa: Componentes de Stripe
376 import { Stripe } from 'stripe';
377
378 // Importa: Componentes de Stripe
379 import { Stripe } from 'stripe';
380
381 // Importa: Componentes de Stripe
382 import { Stripe } from 'stripe';
383
384 // Importa: Componentes de Stripe
385 import { Stripe } from 'stripe';
386
387 // Importa: Componentes de Stripe
388 import { Stripe } from 'stripe';
389
390 // Importa: Componentes de Stripe
391 import { Stripe } from 'stripe';
392
393 // Importa: Componentes de Stripe
394 import { Stripe } from 'stripe';
395
396 // Importa: Componentes de Stripe
397 import { Stripe } from 'stripe';
398
399 // Importa: Componentes de Stripe
400 import { Stripe } from 'stripe';
401
402 // Importa: Componentes de Stripe
403 import { Stripe } from 'stripe';
404
405 // Importa: Componentes de Stripe
406 import { Stripe } from 'stripe';
407
408 // Importa: Componentes de Stripe
409 import { Stripe } from 'stripe';
410
411 // Importa: Componentes de Stripe
412 import { Stripe } from 'stripe';
413
414 // Importa: Componentes de Stripe
415 import { Stripe } from 'stripe';
416
417 // Importa: Componentes de Stripe
418 import { Stripe } from 'stripe';
419
420 // Importa: Componentes de Stripe
421 import { Stripe } from 'stripe';
422
423 // Importa: Componentes de Stripe
424 import { Stripe } from 'stripe';
425
426 // Importa: Componentes de Stripe
427 import { Stripe } from 'stripe';
428
429 // Importa: Componentes de Stripe
430 import { Stripe } from 'stripe';
431
432 // Importa: Componentes de Stripe
433 import { Stripe } from 'stripe';
434
435 // Importa: Componentes de Stripe
436 import { Stripe } from 'stripe';
437
438 // Importa: Componentes de Stripe
439 import { Stripe } from '
```

RESULTADOS:

-PRIMER RESULTADO DE TODO LO QUE PONGO MANUAL:

```
slzr0@DESKTOP-77K2F6E MINGW64 /d/ADSO/TERCER TRIMESTRE/JULIAN SALAZAR/TALLER HERENCIA/01
○ $ dart main.dart
PRIMERA PERSONA
  nombre: Daniel
  edad: 17

*****

PRIMER FUTBOLISTA
  nombre: Daniel
  edad: 17
  equipo: Barcelona
  posicion: MC
  goles: 7
  Titularidad: true

*****

SEGUNDO FUTBOLISTA
  nombre: Miguel
  edad: 17
  equipo: Real Madrid
  posicion: DC
  goles: 3
  Titularidad: false

*****
*****
  nombre: Daniel
  edad: 17
  empresa: Aguas Manizales
  salario: 1068000.0

*****
```

```

PRIMER PROGRAMADOR
nombre: Daniel
edad: 17
empresa: Aguas Manizales
salario: 1068000.0
salario Neto: 950520.0

*****
*****

nombre: Miguel
edad: 17
empresa: Chec
salario: 1068000.0

*****
*****

SEGUNDO PROGRAMADOR
nombre: Miguel
edad: 17
empresa: Chec
salario: 1068000.0
salario Neto: 950520.0

*****
*****

```

-INGRESO DEL PRIMER FUTBOLISTA:

```

Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
1
Ingrese el nombre del futbolista 1
Daniel
Ingrese la edad del futbolista:
17
Ingrese el equipo del futbolista:
Barcelona
Ingrese la posicion del futbolista:
MC
Ingrese la cantidad de goles:
102

```


-INGRESO DEL SEGUNDO FUTBOLISTA:

```
Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
1
Ingrese el nombre del futbolista 1
Miguel
Ingrese la edad del futbolista:
18
Ingrese el equipo del futbolista:
Real Madrid
Ingrese la posicion del futbolista:
DC
Ingrese la cantidad de goles:
7
```

-INGRESO DEL PRIMER PROGRAMADOR:

```
Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
2
Ingrese el nombre del programador:
Daniel
Ingrese la edad del programador:
17
Ingrese la empresa en la que esta el programador:
Aguas Manizales
Ingrese el salario del programador:
1200000
```

-INGRESO DEL SEGUNDO PROGRAMADOR:

```
Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
2
Ingrese el nombre del programador:
Miguel
Ingrese la edad del programador:
18
Ingrese la empresa en la que esta el programador:
Chec
Ingrese el salario del programador:
1200000
```

-MUESTRA DE LOS DATOS DE LOS FUTBOLISTAS AGREGADOS:

```
Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
3
*****
    nombre: Daniel
    edad: 17
    equipo: Barcelona
    posicion: MC
    goles: 102
    Titularidad: true
*****
*****
    nombre: Miguel
    edad: 18
    equipo: Real Madrid
    posicion: DC
    goles: 7
    Titularidad: true
*****
```

-MUESTRA DE LOS DATOS DE LOS PROGRAMADORES AGREGADOS:

```
Menu:
1.Agregar futbolista.
2.Agregar programador.
3.Mostrar a los futbolistas agregados.
4.Mostrar a los programadores agregados.
4
*****
    nombre: Daniel
    edad: 17
    empresa: Aguas Manizales
    salario: 1200000.0
*****
*****
    nombre: Miguel
    edad: 18
    empresa: Chec
    salario: 1200000.0
*****
```

2. Crear un proyecto en Dart y realizar lo siguiente:

A. Construir las clases: Figura (clase padre), Circulo, Cuadrado y Triángulo aplicando los conceptos de herencia y polimorfismo.

B. Se deben crear objetos de cada una de las clases. Tomar un pantallazo de los resultados

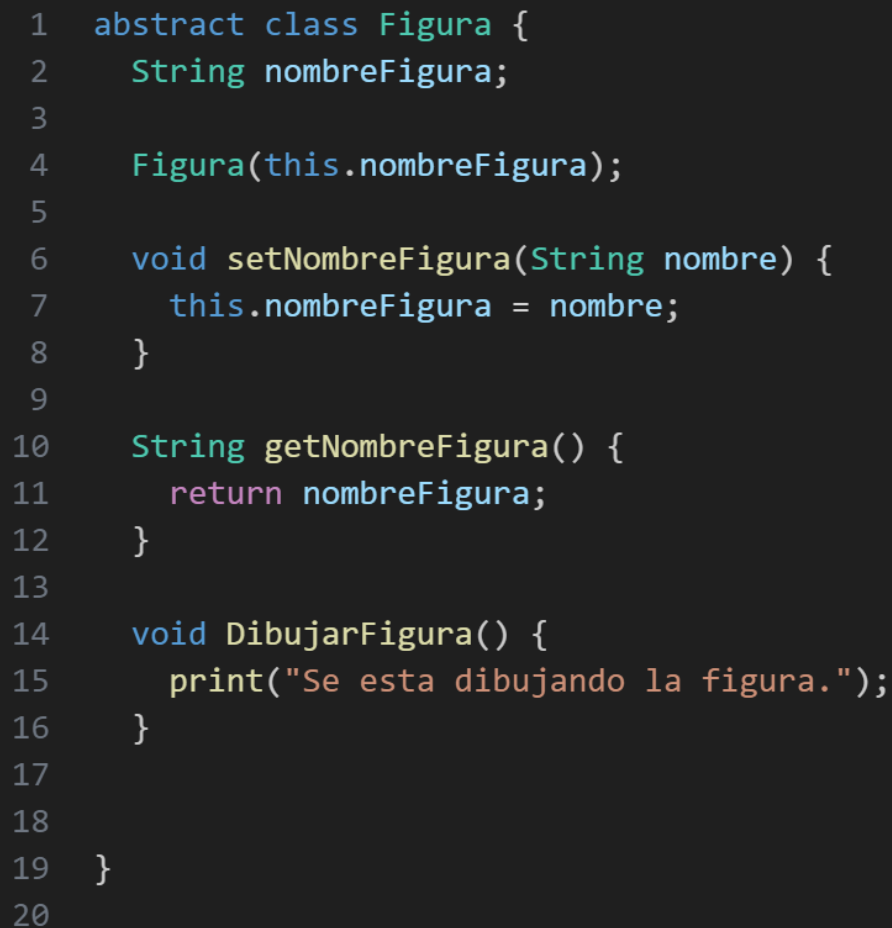
C. Se debe permitir calcular el área de la figura específica.

SOLUCION

En este archivo de las figuras de la parte 2 lo que hice fue crear las páginas “Figura.dart”, “Circulo.dart”, “Cuadrado.dart”, “Triangulo.dart” y “main.dart”.

CODIGOS:

Figura.dart:



```
1  abstract class Figura {
2      String nombreFigura;
3
4      Figura(this.nombreFigura);
5
6      void setNombreFigura(String nombre) {
7          this.nombreFigura = nombre;
8      }
9
10     String getNombreFigura() {
11         return nombreFigura;
12     }
13
14     void DibujarFigura() {
15         print("Se esta dibujando la figura.");
16     }
17
18
19 }
20
```

Circulo.dart:

```
1  import 'Figura.dart';
2
3  class Circulo extends Figura {
4      double radio;
5
6      Circulo(this.radio, String nombreFigura) : super(nombreFigura);
7
8      double CalcularAreaCirculo() {
9          double area;
10         double pi;
11         pi = 3.1416;
12         area = pi * (radio * radio);
13         return area;
14     }
15
16     @override
17     void DibujarFigura() {
18         super.DibujarFigura();
19     }
20
21     @override
22     String getNombreFigura() {
23         return super.getNombreFigura();
24     }
25
26     void mostrarInfoCirculo() {
27         print("""
28             Figura: $nombreFigura
29             Area: ${CalcularAreaCirculo()}
30         """);
31     }
32 }
33
```

Cuadrado.dart:

```
1  import 'Figura.dart';
2
3  class Cuadrado extends Figura {
4      double lado1;
5      double lado2;
6
7      Cuadrado(this.lado1, this.lado2, String nombreFigura) : super(nombreFigura);
8
9      double CalcularAreaCuadrado() {
10         double lado;
11         lado = lado1 * lado2;
12         return lado;
13     }
14
15     @override
16     void DibujarFigura() {
17         super.DibujarFigura();
18     }
19
20     @override
21     String getNombreFigura() {
22         return super.getNombreFigura();
23     }
24
25     void mostrarInfoCuadrado() {
26         print("""
27         Figura: $nombreFigura
28         Area: ${CalcularAreaCuadrado()}
29         """);
30     }
31 }
32
```

Triangulo.dart:

```
1  import 'Figura.dart';
2
3  class Triangulo extends Figura {
4      double base;
5      double altura;
6
7      Triangulo(this.base, this.altura, String nombreFigura) : super(nombreFigura);
8
9      double CalcularAreaTriangulo() {
10         double area;
11         area = (base * altura) / 2;
12         return area;
13     }
14
15     @override
16     void DibujarFigura() {
17         super.DibujarFigura();
18     }
19
20     @override
21     String getNombreFigura() {
22         return super.getNombreFigura();
23     }
24
25     void mostrarInfoTriangulo() {
26         print("""
27         Figura: $nombreFigura
28         Area: ${CalcularAreaTriangulo()}
29         """);
30     }
31 }
32
```

main.dart:

```
1 import 'dart:io';
2
3 import 'Circulo.dart';
4 import 'Cuadrado.dart';
5 import 'Triangulo.dart';
6
7 void main(List<String> args) {
8   /*
9   Circulo circulo01 = Circulo(8);
10  circulo01.CalcularAreaCirculo();
11  circulo01.DibujarFigura();
12
13  print('*' * 50);
14
15  Cuadrado cuadrado01 = Cuadrado(9, 9);
16  cuadrado01.CalcularAreaCuadrado();
17  cuadrado01.DibujarFigura();
18
19  print('*' * 50);
20
21  Triangulo triangulo01 = Triangulo(3, 9);
22  triangulo01.CalcularAreaTriangulo();
23  triangulo01.DibujarFigura();
24  */
25  print('*' * 50);
26
27  List<Circulo> circulo = [];
28  List<Cuadrado> cuadrado = [];
29  List<Triangulo> triangulo = [];
30
31  while (true) {
32    print("Menu:");
33    print("1.Calcular el area de un circulo.-");
34    print("2.Calcular el area de un cuadrado.");
35    print("3.Calcular el area de un Triangulo.");
36    print("4.Salir.");
37
38    int opcion = int.parse(stdin.readLineSync()!);
39
40    switch (opcion) {
41      case 1:
42        print('*' * 30);
43        print("Ingrese el valor del radio del circulo: ");
44        double radio = double.parse(stdin.readLineSync()!);
45
46        /*circulo01.CalcularAreaCirculo();
47        Circulo(radio);*/
48        Circulo circulo1 = Circulo(radio, "Circulo");
49        circulo1.CalcularAreaCirculo();
50        circulo1.DibujarFigura();
51        circulo1.getNombreFigura();
52        circulo.add(circulo1);
53        print('*' * 30);
54        break;
55
56      case 2:
57        print('*' * 30);
58        print("Ingrese el valor del lado 1 del cuadrado: ");
59        double lado1 = double.parse(stdin.readLineSync()!);
60
61        print("Ingrese el valor del lado 2 del cuadrado: ");
62        double lado2 = double.parse(stdin.readLineSync()!);
63
64        /*cuadrado01.CalcularAreaCuadrado();
65        Cuadrado(lado1, lado2);*/
66
67        Cuadrado cuadrado1 = Cuadrado(lado1, lado2, "Cuadrado");
68        cuadrado1.CalcularAreaCuadrado();
69        cuadrado1.DibujarFigura();
70        cuadrado1.getNombreFigura();
71        cuadrado.add(cuadrado1);
72        print('*' * 30);
73        break;
74
75      case 3:
76        print('*' * 30);
77        print("Ingrese el valor de la base del triangulo: ");
78        double base = double.parse(stdin.readLineSync()!);
79
80        print("Ingrese el valor de la altura del triangulo: ");
81        double altura = double.parse(stdin.readLineSync()!);
82
83        /*triangulo01.CalcularAreaTriangulo();
84        Triangulo(base, altura);*/
85        Triangulo triangulo1 = Triangulo(base, altura, "Triangulo");
86        triangulo1.CalcularAreaTriangulo();
87        triangulo1.DibujarFigura();
88        triangulo1.getNombreFigura();
89        triangulo.add(triangulo1);
90        print('*' * 30);
91        break;
92
93      case 4:
94        print("Saliendo de la pagina.");
95        exit(0);
96      default:
97        print("Esta opcion no existe, vuelva a intentarlo.");
98    }
99  }
100 }
101
```


TERMINAL:

-INGRESO Y RESULTADO DEL CIRCULO:

```
Menu:
1.Calcular el area de un circulo.
2.Calcular el area de un cuadrado.
3.Calcular el area de un Triangulo.
4.Salir.
1
*****
Ingrese el valor del radio del circulo:
10
El area del circulo es: 314.15999999999997
Se esta dibujando la figura...
*****
```

-INGRESO Y RESULTADO DEL CUADRADO:

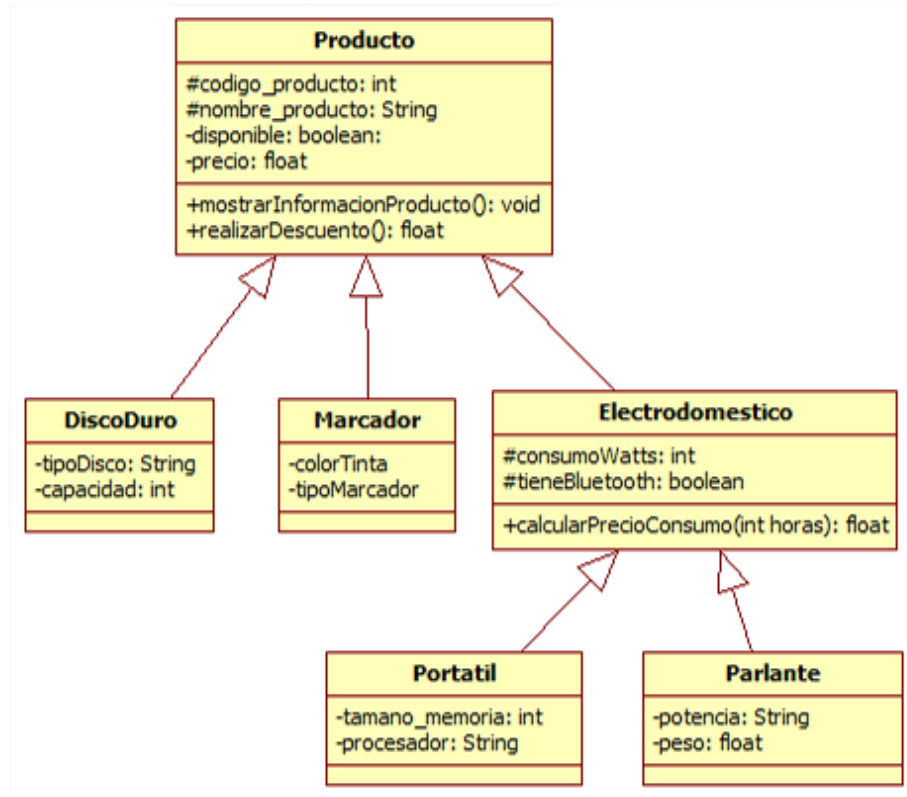
```
Menu:
1.Calcular el area de un circulo.
2.Calcular el area de un cuadrado.
3.Calcular el area de un Triangulo.
4.Salir.
2
*****
Ingrese el valor del lado 1 del cuadrado:
8
Ingrese el valor del lado 2 del cuadrado:
8
El area del cuadrado es: 64.0
Se esta dibujando la figura...
*****
```

-INGRESO Y RESULTADO DEL TRIANGULO:

```
Menu:
1.Calcular el area de un circulo.
2.Calcular el area de un cuadrado.
3.Calcular el area de un Triangulo.
4.Salir.
3
*****
Ingrese el valor de la base del triangulo:
6
Ingrese el valor de la altura del triangulo:
10
El area del triangulo es: 30.0
Se esta dibujando la figura...
*****
```

GUIA TALLER HERENCIA Y POLIMORFISMO EN DART – 02

1. Crear un proyecto en Dart y seguir las siguientes instrucciones, tomando como base el siguiente diagrama de clases:



A. Construir las 6 clases en Dart (utilizando Visual Studio Code) tomando como base el diagrama de clases anterior y teniendo en cuenta la secuencia de herencia manejada. Deben crearse todos los métodos SET y GET de cada clase.

B. La clase **Producto** deberá crearse como clase abstracta, al igual que el método `realizarDescuento()`

C. Para llamar el constructor de la clase **Producto** desde la clase **DiscoDuro** y **Marcador**, se deberá hacer utilizando el método `super()`

D. Para llamar el constructor de la clase **Producto** desde la clase **Electrodomestico**, se deberán setear directamente los atributos: `código_producto` y `nombre_producto` de la clase padre, y hacer el llamado de un constructor de la clase padre que reciba solamente los valores de los campos privados: `disponible` y `precio`.

E. Sobrecribir el método realizarDescuento() para que quede de la siguiente manera:

I. El descuento de los discos duros será del 20%

II. El descuento de los Marcadores será del 10%

III. El descuento de los Electrodomésticos será del 30%

F. El método mostrarInformacionProducto() de la clase Producto, mostrará todos los atributos de dicha clase.

G. El método calcularPrecioConsumo() de la clase Electrodoméstico, calculará el precio a pagar según el consumo de watts y la cantidad de horas.

H. ¿Intentar crear una instancia (objeto) de la clase Producto y decir si se puede o no y por qué?

I. Al ejecutar el programa, se deberá mostrar un menú como el siguiente:

Elige una opción:

1) Crear Disco Duro

2) Crear Marcador

3) Crear Portátil

4) Crear Parlante

5) Vender Disco Duro

6) Vender Marcador

7) Vender Portátil

8) Vender Parlante

9) Calcular precio consumo Portátil

10) Calcular precio consumo Parlante

11) Salir

J. Si se elige entre la opción 1 y 4, se deberán crear las instancias (objetos) según la clase que corresponda.

K. Si se elige entre la opción 5 y 8, se deberá llamar al método realizarDescuento() según el producto escogido y mostrar toda la información del producto vendido.

L. Si se elige la opción 9 o 10, se deberá mostrar el valor que se pagaría por el consumo de Watts del electrodoméstico escogido según la cantidad de horas ingresadas.

M. Si se elige la opción 11, se deberá terminar con la ejecución del programa.

CODIGO:

```

1  #!/usr/bin/perl
2
3  use strict;
4  use warnings;
5
6  my $VERSION = "1.0.0";
7  my $AUTHOR = "John Doe";
8  my $COPYRIGHT = "Copyright 2023 John Doe";
9
10 my $SYNOPSIS = "Usage: perl script.pl [options]";
11 my $DESCRIPTION = "A script to process data files and generate reports.";
12
13 my $USAGE = "
14     perl script.pl [options]
15
16     Options:
17     -h, --help            Display this help message.
18     -v, --version         Display the version number.
19     -i, --input FILENAME Input file name.
20     -o, --output FILENAME Output file name.
21     -f, --format FORMAT   Output format (text, json, csv).
22     -s, --size SIZE       File size limit in MB.
23     -d, --debug           Enable debug mode.
24     -q, --quiet           Suppress non-error messages.
25     -n, --no-color        Disable color output.
26     -e, --error FILENAME  Log error messages to FILENAME.
27     -c, --config FILENAME Load configuration from FILENAME.
28     -m, --mode MODE       Operation mode (analyze, report, export).
29     -t, --timeout SECS    Timeout in seconds for network requests.
30     -u, --url URL          Remote resource URL.
31     -a, --auth AUTH       Authentication token.
32     -k, --key KEY          API key.
33     -p, --port PORT       Remote service port.
34     -H, --header HEADER   Custom HTTP header.
35     -X, --method METHOD     HTTP method (GET, POST, PUT, DELETE).
36     -P, --proxy PROXY      Proxy server address.
37     -S, --ssl              Enable SSL/TLS.
38     -C, --cert CERT        Client certificate path.
39     -K, --key KEY           Private key path.
40     -E, --engine ENGINE     Processing engine (cpu, gpu).
41     -M, --memory MB        Memory allocation in MB.
42     -T, --threads THREADS  Number of threads.
43     -B, --batch SIZE        Batch size for processing.
44     -I, --interval SECS    Interval between operations.
45     -R, --retry SECS        Retry delay in seconds.
46     -L, --limit COUNT       Maximum number of items.
47     -F, --filter FILTER     Filter expression.
48     -G, --group GROUP       Group identifier.
49     -O, --order ORDER       Sort order (asc, desc).
50     -S, --sort SORT         Sort criteria.
51     -A, --agg AGG           Aggregation function.
52     -C, --cond COND         Conditional expression.
53     -J, --join JOIN         Join type (inner, left, right, full).
54     -P, --part PART        Partition number.
55     -T, --tbl TAB           Table name.
56     -C, --col COL           Column name.
57     -F, --fld FLD           Field name.
58     -I, --idx IDX           Index name.
59     -P, --prt PRG           Program name.
60     -M, --mod MOD           Module name.
61     -P, --pkg PKG           Package name.
62     -N, --ns NAMESPACE     Namespace.
63     -U, --url URL           URL path.
64     -Q, --qry QUERY         SQL query.
65     -D, --db DB             Database name.
66     -U, --usr USER          Username.
67     -P, --pwd PASS          Password.
68     -H, --host HOST         Hostname.
69     -P, --port PORT         Port number.
70     -S, --ssl SSL           SSL certificate.
71     -C, --cert CERT         Client certificate.
72     -K, --key KEY           Private key.
73     -E, --engine ENGINE     Engine type.
74     -M, --memory MB         Memory size.
75     -T, --threads THREADS   Thread count.
76     -B, --batch SIZE        Batch size.
77     -I, --interval SECS     Interval time.
78     -R, --retry SECS        Retry delay.
79     -L, --limit COUNT       Limit count.
80     -F, --filter FILTER     Filter string.
81     -G, --group GROUP       Group name.
82     -O, --order ORDER       Order type.
83     -S, --sort SORT         Sort field.
84     -A, --agg AGG           Aggregation.
85     -C, --cond COND         Condition.
86     -J, --join JOIN         Join type.
87     -P, --part PART        Partition.
88     -T, --tbl TAB           Table.
89     -C, --col COL           Column.
90     -F, --fld FLD           Field.
91     -I, --idx IDX           Index.
92     -P, --prt PRG           Program.
93     -M, --mod MOD           Module.
94     -P, --pkg PKG           Package.
95     -N, --ns NAMESPACE     Namespace.
96     -U, --url URL           URL.
97     -Q, --qry QUERY         Query.
98     -D, --db DB             Database.
99     -U, --usr USER          User.
100    -P, --pwd PASS           Password.
101    -H, --host HOST          Host.
102    -P, --port PORT          Port.
103    -S, --ssl SSL            SSL.
104    -C, --cert CERT         Certificate.
105    -K, --key KEY            Key.
106    -E, --engine ENGINE      Engine.
107    -M, --memory MB         Memory.
108    -T, --threads THREADS    Threads.
109    -B, --batch SIZE        Batch.
110    -I, --interval SECS     Interval.
111    -R, --retry SECS        Retry.
112    -L, --limit COUNT       Limit.
113    -F, --filter FILTER     Filter.
114    -G, --group GROUP       Group.
115    -O, --order ORDER       Order.
116    -S, --sort SORT         Sort.
117    -A, --agg AGG           Agg.
118    -C, --cond COND         Cond.
119    -J, --join JOIN         Join.
120    -P, --part PART        Part.
121    -T, --tbl TAB           Table.
122    -C, --col COL           Col.
123    -F, --fld FLD           Fld.
124    -I, --idx IDX           Idx.
125    -P, --prt PRG           Prt.
126    -M, --mod MOD           Mod.
127    -P, --pkg PKG           Pkg.
128    -N, --ns NAMESPACE     Ns.
129    -U, --url URL           Url.
130    -Q, --qry QUERY         Qry.
131    -D, --db DB             Db.
132    -U, --usr USER          User.
133    -P, --pwd PASS           Pass.
134    -H, --host HOST         Host.
135    -P, --port PORT         Port.
136    -S, --ssl SSL           Ssl.
137    -C, --cert CERT         Cert.
138    -K, --key KEY           Key.
139    -E, --engine ENGINE     Engine.
140    -M, --memory MB         Memory.
141    -T, --threads THREADS   Threads.
142    -B, --batch SIZE        Batch.
143    -I, --interval SECS     Interval.
144    -R, --retry SECS        Retry.
145    -L, --limit COUNT       Limit.
146    -F, --filter FILTER     Filter.
147    -G, --group GROUP       Group.
148    -O, --order ORDER       Order.
149    -S, --sort SORT         Sort.
150    -A, --agg AGG           Agg.
151    -C, --cond COND         Cond.
152    -J, --join JOIN         Join.
153    -P, --part PART        Part.
154    -T, --tbl TAB           Table.
155    -C, --col COL           Column.
156    -F, --fld FLD           Field.
157    -I, --idx IDX           Index.
158    -P, --prt PRG           Program.
159    -M, --mod MOD           Module.
160    -P, --pkg PKG           Package.
161    -N, --ns NAMESPACE     Namespace.
162    -U, --url URL           URL.
163    -Q, --qry QUERY         Query.
164    -D, --db DB             Database.
165    -U, --usr USER          Username.
166    -P, --pwd PASS           Password.
167    -H, --host HOST         Hostname.
168    -P, --port PORT         Port.
169    -S, --ssl SSL           SSL/TLS.
170    -C, --cert CERT         Client certificate.
171    -K, --key KEY           Private key.
172    -E, --engine ENGINE     Engine.
173    -M, --memory MB         Memory.
174    -T, --threads THREADS   Thread count.
175    -B, --batch SIZE        Batch size.
176    -I, --interval SECS     Interval.
177    -R, --retry SECS        Retry delay.
178    -L, --limit COUNT       Limit.
179    -F, --filter FILTER     Filter.
180    -G, --group GROUP       Group.
181    -O, --order ORDER       Order.
182    -S, --sort SORT         Sort.
183    -A, --agg AGG           Aggregation.
184    -C, --cond COND         Condition.
185    -J, --join JOIN         Join.
186    -P, --part PART        Partition.
187    -T, --tbl TAB           Table.
188    -C, --col COL           Column.
189    -F, --fld FLD           Field.
190    -I, --idx IDX           Index.
191    -P, --prt PRG           Program.
192    -M, --mod MOD           Module.
193    -P, --pkg PKG           Package.
194    -N, --ns NAMESPACE     Namespace.
195    -U, --url URL           URL.
196    -Q, --qry QUERY         Query.
197    -D, --db DB             Database.
198    -U, --usr USER          Username.
199    -P, --pwd PASS           Password.
200    -H, --host HOST         Hostname.
201    -P, --port PORT         Port.
202    -S, --ssl SSL           SSL/TLS.
203    -C, --cert CERT         Client certificate.
204    -K, --key KEY           Private key.
205    -E, --engine ENGINE     Engine.
206    -M, --memory MB         Memory.
207    -T, --threads THREADS   Thread count.
208    -B, --batch SIZE        Batch size.
209    -I, --interval SECS     Interval.
210    -R, --retry SECS        Retry delay.
211    -L, --limit COUNT       Limit.
212    -F, --filter FILTER     Filter.
213    -G, --group GROUP       Group.
214    -O, --order ORDER       Order.
215    -S, --sort SORT         Sort.
216    -A, --agg AGG           Aggregation.
217    -C, --cond COND         Condition.
218    -J, --join JOIN         Join.
219    -P, --part PART        Partition.
220    -T, --tbl TAB           Table.
221    -C, --col COL           Column.
222    -F, --fld FLD           Field.
223    -I, --idx IDX           Index.
224    -P, --prt PRG           Program.
225    -M, --mod MOD           Module.
226    -P, --pkg PKG           Package.
227    -N, --ns NAMESPACE     Namespace.
228    -U, --url URL           URL.
229    -Q, --qry QUERY         Query.
230    -D, --db DB             Database.
231    -U, --usr USER          Username.
232    -P, --pwd PASS           Password.
233    -H, --host HOST         Hostname.
234    -P, --port PORT         Port.
235    -S, --ssl SSL           SSL/TLS.
236    -C, --cert CERT         Client certificate.
237    -K, --key KEY           Private key.
238    -E, --engine ENGINE     Engine.
239    -M, --memory MB         Memory.
240    -T, --threads THREADS   Thread count.
241    -B, --batch SIZE        Batch size.
242    -I, --interval SECS     Interval.
243    -R, --retry SECS        Retry delay.
244    -L, --limit COUNT       Limit.
245    -F, --filter FILTER     Filter.
246    -G, --group GROUP       Group.
247    -O, --order ORDER       Order.
248    -S, --sort SORT         Sort.
249    -A, --agg AGG           Aggregation.
250    -C, --cond COND         Condition.
251    -J, --join JOIN         Join.
252    -P, --part PART        Partition.
253    -T, --tbl TAB           Table.
254    -C, --col COL           Column.
255    -F, --fld FLD           Field.
256    -I, --idx IDX           Index.
257    -P, --prt PRG           Program.
258    -M, --mod MOD           Module.
259    -P, --pkg PKG           Package.
260    -N, --ns NAMESPACE     Namespace.
261    -U, --url URL           URL.
262    -Q, --qry QUERY         Query.
263    -D, --db DB             Database.
264    -U, --usr USER          Username.
265    -P, --pwd PASS           Password.
266    -H, --host HOST         Hostname.
267    -P, --port PORT         Port.
268    -S, --ssl SSL           SSL/TLS.
269    -C, --cert CERT         Client certificate.
270    -K, --key KEY           Private key.
271    -E, --engine ENGINE     Engine.
272    -M, --memory MB         Memory.
273    -T, --threads THREADS   Thread count.
274    -B, --batch SIZE        Batch size.
275    -I, --interval SECS     Interval.
276    -R, --retry SECS        Retry delay.
277    -L, --limit COUNT       Limit.
278    -F, --filter FILTER     Filter.
279    -G, --group GROUP       Group.
280    -O, --order ORDER       Order.
281    -S, --sort SORT         Sort.
282    -A, --agg AGG           Aggregation.
283    -C, --cond COND         Condition.
284    -J, --join JOIN         Join.
285    -P, --part PART        Partition.
286    -T, --tbl TAB           Table.
287
```

TERMINAL:

-CREAR DISCO DURO:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
1
```

```
*****
INGRESE LOS SIGUIENTES DATOS PARA CREAR EL DISCO DURO:
Ingrese el tipo de Disco duro:
Disco
Ingrese la capacidad del Disco duro:
100
Ingrese el codigo del Disco duro:
10
Ingrese el nombre del Disco duro:
Daniel
Ingrese el precio del Disco duro:
120000
```

```
*****
Codigo: 10
Nombre: Daniel
Disponibilidad: true
Precio: 120000.0
*****
```

-CREAR MARCADOR:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
2
```

```

*****
INGRESE LOS SIGUIENTES DATOS PARA CREAR EL MARCADOR:
Ingrese el color de la tinta:
Negro
Ingrese el tipo de marcador:
Norma
Ingrese el codigo del marcador:
10
Ingrese el nombre del marcador :
Daniel
Ingrese el precio del Marcador:
12000

```

```

*****
Codigo: 10
Nombre: Daniel
Disponibilidad: true
Precio: 12000.0
*****

```

-CREAR PORTATIL:

```

MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
3

```

```

*****
INGRESE LOS SIGUIENTES DATOS PARA INGRESAR EL PORTATIL:
Ingrese el tamaño de la memoria del portatil:
100
Ingrese el procesador del portatil:
DANIEL
Ingrese el consumo de Watts del portatil:
100
Ingrese el precio del portatil:
230000

```

```

*****
Codigo: 0
Nombre:
Disponibilidad: true
Precio: 230000.0
*****

```

-CREAR PARLANTE:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
4
```

```
*****
INGRESE LOS SIGUIENTES DATOS PARA INGRESAR EL PARLANTE:
```

Ingrese la potencia del parlante:

100

Ingrese el peso del parlante:

12

Ingrese el consumo de Watts del parlante:

50

Ingrese el precio del parlante:

65000

```
*****
Codigo: 0
```

Nombre:

Disponibilidad: true

Precio: 65000.0

```
*****
```

-VENDER DISCO DURO:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
5
```

```
*****
DISCOS DUROS DISPONIBLES:
```

No esta disponible el producto.

```
*****
```

-VENDER MARCADOR:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
6
```

```
*****
MARCADORES DISPONIBLES:
No esta disponible el producto.
*****
```

-VENDER PORTATIL:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
7
```

```
*****
PORTATILES DISPONIBLES
No esta disponible el producto.
*****
```


-VENDER PARLANTE:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil
10.Calcular precio consumo Parlante
11.Salir.
8
```

```
*****
PARLANTES DISPONIBLES:
No esta disponible el producto.
*****
```

-CALCULAR PRECIO CONSUMO PORTATIL:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
9
```

```
*****
CONSUMO PORTATIL:
Cuanto horas uso el dispositivo:
10
```

-CALCULAR PRECIO CONSUMO PARLANTE:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
10
```

```
*****
CONSUMO PARLANTE:
Cuanto horas uso el dispositivo:
20
```

-SALIR:

```
MENU:
1.Crear Disco Duro.
2.Crear Marcador.
3.Crear Portátil.
4.Crear Parlante.
5.Vender Disco Duro.
6.Vender Marcador.
7.Vender Portátil.
8.Vender Parlante.
9.Calcular precio consumo Portátil.
10.Calcular precio consumo Parlante.
11.Salir.
11
```

```
*****
Saliendo de la pagina.
*****
```

```
slzr@DESKTOP-77K2F6E MINGW64 /d/ADSO/TERCER TRIMESTRE/JULIAN SALAZAR/TALLER HERENCIA/02
```

```
$
```

DANIEL SALAZAR LOAIZA – ADSO 2873707